

# RESUMO PARA A SEGUNDA PROVA DE LINGUAGENS FORMAIS

## → CÁLCULO DO FIRST(X)

É o conjunto de terminais que podem iniciar alguma derivação a partir de X.

① Se X é terminal,  $FIRST(X) = \{X\}$

② Se  $X \rightarrow \epsilon$  é uma produção, então  $\epsilon \in FIRST(X)$

③ Se  $X \rightarrow Y_1 Y_2 \dots Y_n$  é uma produção, para cada símbolo  $Y_i$ , adicione todos os símbolos de  $FIRST(Y_i)$ , exceto  $\epsilon$ , ao conjunto  $FIRST(X)$ . Se  $FIRST(Y_i)$  contém  $\epsilon$ , continue analisando  $Y_{i+1}$ , caso contrário, pare. Se  $\epsilon \in FIRST(Y_i)$  para todos  $i = 1$  até  $n$ , então  $\epsilon \in FIRST(X)$

## → CÁLCULO DO FOLLOW(X)

É o conjunto de terminais que podem aparecer imediatamente após X em alguma derivação válida a partir do símbolo inicial da gramática.

① Se X é a produção inicial, então  $\$ \in FOLLOW(X)$

② Para qualquer produção na forma  $A \rightarrow \alpha B \beta$ ,  $FIRST(\beta)$ , exceto  $\epsilon$ , é adicionado ao  $FOLLOW(B)$ . Destaca-se que  $\beta$  é a concatenação de tudo o que vem depois de B.

③ Para qualquer produção  $A \rightarrow \alpha B$  ou  $A \rightarrow \alpha B \beta$ , sendo  $\beta \rightarrow \epsilon$ , então  $FOLLOW(A) \subseteq FOLLOW(B)$

## → EXEMPLO DE FIRST E FOLLOW

Considerando a gramática:

$E \rightarrow TE'$	(, id	\$, )
$E' \rightarrow +TE' \mid \epsilon$	+, $\epsilon$	\$, )
$T \rightarrow FT'$	(, id	+, \$, )
$T' \rightarrow *FT' \mid \epsilon$	*, $\epsilon$	+, \$, )
$F \rightarrow (E) \mid id$	(, id	*, +, \$, )
	FIRST	FOLLOW

## → EXEMPLO DE TABELA PREDITIVA

Considerando a gramática acima:

	id	+	*	(	)	\$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$			$E' \rightarrow \epsilon$	$E' \rightarrow \epsilon$
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'		$T' \rightarrow \epsilon$	$T' \rightarrow *FT'$		$T' \rightarrow \epsilon$	$T' \rightarrow \epsilon$
F	$F \rightarrow id$			$F \rightarrow (E)$		

## → ANALISADOR DESCENDENTE TABULAR PREDITIVO

Precisase:

① Ter uma gramática livre de contexto fatorada e sem recursão.

② Calcular os conjuntos FIRST e FOLLOW.

③ Construir a tabela preditiva, onde as linhas são os não terminais e as colunas são os terminais mais o \$.

↳ Para cada produção  $A \rightarrow \alpha$  da gramática, faça:

③A Para cada terminal a de  $FIRST(\alpha)$  com  $a \neq \epsilon$ , coloque  $A \rightarrow \alpha$  em  $M[A, a]$

③B Se  $\epsilon \in FIRST(\alpha)$ , então para cada  $b \in FOLLOW(A)$ , coloque a produção  $A \rightarrow \alpha$  em  $M[A, b]$

O analisador inicializa a pilha com \$ e a regra inicial. Ele analisa o símbolo no topo da pilha, X, e o símbolo corrente na entrada, a. Segue-se assim:

① Se  $X = a = \$$ , sucesso

② Se  $X = a \neq \$$ , remove X da pilha e avança entrada.

③ Se X é não terminal, X é substituído na pilha pelos símbolos do lado direito da produção (em ordem inversa)

## → ANALISADOR DESCENDENTE RECURSIVO COM RETROCESS.

Se  $S \rightarrow aB \mid aC$

$B \rightarrow ab$

$C \rightarrow aC$ , então:

```
int S() {
    if (token_atual == TK-a) {
        le_token();
        marca_posicao();
        if (B()) {
            return TRUE;
        } else {
            restaura_posicao();
            if (C()) {
                return TRUE;
            } else {
                return FALSE;
            }
        }
    }
    return FALSE;
}
```

return FALSE;

↑ FATORAÇÃO  
Se  $A \rightarrow \alpha B \mid \gamma$ , então:  
 $A \rightarrow \alpha A'$   
 $A \rightarrow \beta \mid \gamma$   
Se for não-determinismo indireto, faça substituição primeiro.



## → REMOÇÃO DA RECURSÃO À ESQUERDA

$A \rightarrow A\alpha_1 \mid A\alpha_2 \mid B_1 \mid B_2$

↓

$A \rightarrow B_1 A' \mid B_2 A'$

$A' \rightarrow \alpha_1 A' \mid \alpha_2 A'$

} recursão direta

$A \rightarrow Ba \mid c$

$B \rightarrow Ab \mid d$

↓

Ordem = A, B

↓

↳ nada para mod. aqui

Em B fazemos uma substituição

$B \rightarrow Bab \mid cb \mid d$

Agora temos uma recursão direta

} recursão indireta

## → ANALISADORES SLR, LR(0) E LR(1)

Constrói-se a coleção de itens, constrói-se a tabela e faz-se o parsing usando a pilha.

## → EXEMPLO

$I_0 = \{ [S' \rightarrow \cdot S, \$],$   
 $[S \rightarrow \cdot a, \$],$   
 $[S \rightarrow \cdot (L), \$] \}$

$I_1 = \text{Goto}(I_0, S) = \{ [S' \rightarrow S \cdot, \$] \}$

$I_2 = \text{Goto}(I_0, a) = \{ [S \rightarrow a \cdot, \$] \}$

$I_3 = \text{Goto}(I_0, () = \{ \dots \}$

$I_4 = \text{goto}(I_3, L) = \{ [S \rightarrow (L) \cdot, \$] \}$

$I_5 = \text{goto}(I_3, S) = \{ [L \rightarrow S \cdot, (']],$   
 $[L \rightarrow S \cdot, ('] \}$

$I_6 = \text{goto}(I_3, a = \{ [S \rightarrow a \cdot, ('] \})$

$I_7 = \text{goto}(I_4, ') = \{ [S \rightarrow (L) \cdot, \$] \}$

$L \rightarrow S$

$L \rightarrow S, L$

$S \rightarrow a$

$S \rightarrow (L)$

## → CONSTRUÇÃO DE ITENS LR(0)

Cria-se o item inicial  $I_0 = \text{closure}([S' \rightarrow \cdot S])$ . O  $\text{closure}(I)$  fecha um conjunto de itens. Para todo item no conjunto  $I$  do tipo  $A \rightarrow \alpha \cdot B\beta$ , se o ponto está antes de um não terminal  $B$ , então cada produção de  $B$  é adicionada ao conjunto na forma  $B \rightarrow \cdot \gamma$ . Por fim, aplica  $\text{goto}(I, X)$ , onde para cada tipo de item  $A \rightarrow \alpha \cdot X\beta$  em  $I$ , produz o item  $A \rightarrow \alpha X \cdot \beta$ . Depois, aplica  $\text{closure}(I)$  no resultado.

## → CONSTRUÇÃO DE ITENS LR(1)

Cria-se o item inicial  $I_0 = \text{closure}([S' \rightarrow \cdot S, \$])$ . No  $\text{closure}(I)$ , para cada item  $[A \rightarrow \alpha \cdot B\beta, a]$ , adicionamos as produções  $B \rightarrow \cdot \gamma$  e o lookahead é o  $\text{FIRST}(B\beta)$ , ou seja, da produção passada para o  $\text{closure}(I)$ . Aplica-se o  $\text{goto}(I, X)$ .

e.g.  $\text{Goto}(I_0, '() = \text{closure}([S \rightarrow (\cdot L), \$])$

o lookahead vem sempre =  $\{ [S \rightarrow (\cdot L), \$], \dots \}$  lookahead da produção do  $\text{closure}$   $[L \rightarrow \cdot S, (']$ , vindo do  $\text{closure}(I)$

$[L \rightarrow \cdot S, (']$ ,  $[S \rightarrow a \cdot, (']$  ao fazer  $\text{closure}$  do  $S$ .

## → CONSTRUÇÃO DA TABELA LR(0) e SLR

Usa itens LR(0). Seja  $C = I_0, I_1, \dots, I_n$ , os estados do analisador serão  $0, 1, \dots, n$ . Para cada estado  $i$ , faça:

① Se  $A \rightarrow \alpha \cdot a\beta$  estão em  $I_i$  e  $\text{goto}(I_i, a) = I_j$ , então  $\text{TAB}[i, a] = e_j$ .

② Se  $A \rightarrow \alpha \cdot$  está em  $I_i$ , então  $\text{TAB}[i, a] = r_n$ , sendo  $n$  o número da produção  $A \rightarrow \alpha$  para todo 'a' em  $\text{Follow}(A)$ , se for SLR, e para toda linha se for LR(0).

③ Se  $S' \rightarrow S \cdot$  está em  $I_i$ , então  $\text{TAB}[i, \$] = \text{ACEITA}$

④ Se  $\text{goto}(I_i, A) = I_j$ , então  $\text{TRANSIÇÃO}(I_i, A) = j$

## → CONSTRUÇÃO DA TABELA LR(1)

Para cada estado  $i$ :

① Se  $[A \rightarrow \alpha \cdot a\beta, b]$  está em  $I_i$  e  $\text{goto}(I_i, a) = I_j$ , então  $\text{TAB}[i, a] = e_j$

② Se  $[A \rightarrow \alpha \cdot, a]$  está em  $I_i$ ,  $A \neq S'$ , então  $\text{TAB}[i, a] = r_n$ , sendo  $n$  o número da produção  $A \rightarrow \alpha$ .

③ Se  $[S' \rightarrow S \cdot, \$]$  está em  $I_i$ , então faça  $\text{TAB}[i, \$] = \text{ACEITA}$

④ Se  $\text{goto}(I_i, A) = I_j$ , então  $\text{TRANSIÇÃO}(I_i, A) = j$ .

No LR(1), a redução é na coluna do lookahead, no LR(0) é em toda a linha e no SLR(1) é no Follow da produção

	a	(	)	,	\$	S	L
0	e2	e3				1	
1					AC		
2					r1		
3	e6					5	4
4			e7				
5			r4				
6			r1	r1			
7					r2		